VEHICLE-PEDESTRIAN INTERACTION USING NATURALISTIC DRIVING VIDEO THROUGH TRACTOGRAPHY OF RELATIVE POSITIONS AND PEDESTRIAN POSE ESTIMATION

A Thesis

Submitted to the Faculty

of

Purdue University

by

Rifat M. Mueid

In Partial Fulfillment of the

Requirements for the Degree

of

Master of Science in Electrical and Computer Engineering

May 2017

Purdue University

Indianapolis, Indiana

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF THESIS APPROVAL

Dr. Lauren Christopher, Chair

Department of Electrical and Computer Engineering

Dr. Yaobin Chen

Department of Electrical and Computer Engineering

Dr. Paul Salama

Department of Electrical and Computer Engineering

Approved by:

Dr. Brian King

Head of the Departmental Graduate Program

I would like to dedicate this work to my parents Md. Abdul Jalil and Shahnaj Begum. I am grateful for your love and support.

ACKNOWLEDGMENTS

I would like to express my heartiest gratitude to my fantastic supervisor Prof. Lauren Christopher, who has mentored and guided me throughout my research work. She has been a very supportive and inspiring mentor and has played an all-important role in making my research experience here a fulfilling one. I am glad that I had the opportunity to work under her and have an enlightening experience.

I would like to thank Prof. Yaobin Chen and Prof. Paul Salama for serving in my thesis committee. I would also like to thank Dr. Renran Tian for collaborating in the project. He has always encouraged me to excel and helped me throughout the project.

Transportation Active Safety Institute (TASI) of IUPUI have provided the naturalistic driving dataset that have been used in the project. They also provided the mannequin crash dataset that have been used to validate the research. I appreciate this support from TASI. Finally, I would like to thank Samsung Global Research Outreach (GRO) for sponsoring the research.

TABLE OF CONTENTS

Page

LIST OF FIGURES

Figure Page

ABBREVIATIONS

PCS	Pre-Collision System
TASI	Transportation Active Safety Institute
IUPUI	Indiana University-Purdue University Indianapolis
CIB	Crash Imminent Braking
FoE	Focus of Expansion
HOG	Histogram of Oriented Gradient
SVM	Support Vector Machine
LIDAR	Light Detection and Ranging

ABSTRACT

Author: Mueid, Rifat M. MSECE
Institute: Purdue University
Degree Received: May 2017
Title: Vehicle-Pedestrian Interaction using Naturalistic Driving Video through Tractography of Relative Positions and Pedestrian Pose Estimation.
Major Professor: Lauren A. Christopher.

Research on robust Pre-Collision Systems (PCS) requires new techniques that will allow a better understanding of the vehicle-pedestrian dynamic relationship, and which can predict pedestrian future movements. Our research analyzed videos from the Transportation Active Safety Institute (TASI) 110-Car naturalistic driving dataset to extract two dynamic pedestrian semantic features. The dataset consists of videos recorded with forward facing cameras from 110 cars over a year in all weather and illumination conditions. This research focuses on the potential-conflict situations where a collision may happen if no avoidance action is taken from driver or pedestrian. We have used 1000 such 15 seconds videos to find vehicle-pedestrian relative dynamic trajectories and pose of pedestrians. Adaptive structural local appearance model and particle filter methods have been implemented and modified to track the pedestrians more accurately. We have developed new algorithm to compute Focus of Expansion (FoE) automatically. Automatically detected FoE height data have a correlation of 0.98 with the carefully clicked human data. We have obtained correct tractography results for over 82% of the videos. For pose estimation, we have used flexible mixture model for capturing co-occurrence between pedestrian body segments. Based on existing single-frame human pose estimation model, we have introduced Kalman filtering and temporal movement reduction techniques to make stable stick-figure videos of the pedestrian dynamic motion. We were able to reduce frame to frame

pixel offset by 86% compared to the single frame method. These tractographs and pose estimation data were used as features to train a neural network for classifying 'potential conflict' and 'no potential conflict' situations. The training of the network achieved 91.2% true label accuracy, and 8.8% false level accuracy. Finally, the trained network was used to assess the probability of collision over time for the 15 seconds videos which generates a spike when there is a 'potential conflict' situation. We have also tested our method with TASI mannequin crash data. With the crash data we were able to get a danger spike for 70% of the videos. The research enables new analysis on potential-conflict pedestrian cases with 2D tractography data and stick-figure pose representation of pedestrians, which provides significant insight on the vehicle-pedestrian dynamics that are critical for safe autonomous driving and transportation safety innovations.

1. INTRODUCTION

Vehicle (driver)-pedestrian interaction is a very important aspect for transportation safety, especially as driving becomes more autonomous. Current systems operate mainly using Crash Imminent Braking (CIB) where the brakes are only applied at the last minute to avoid collisions. As driving becomes more autonomous, earlier actions by the vehicle must be developed in a more comprehensive way to avoid getting into the CIB situations. The dynamic behavior of the pedestrian in traffic can indicate whether the pedestrian is aware or unaware of the oncoming vehicle. Pedestrians also have a negotiation strategy for crossing traffic which is dynamic in nature, and the pedestrian pose (waving vehicle ahead, running, starting and stopping) can indicate to the vehicle important information.

This research has been built on the extensive database of naturalistic driving data taken in recent years at Indiana by Transportation Active Safety Institute (TASI) in Indiana University-Purdue University Indianapolis (IUPUI). This study collected continuous video using forward-facing camera and synchronized other vehicle data from 110 cars for over 1 year of driving. The overall dataset includes video, GPS, accelerometer, vehicle velocity, time, and other information. During the preliminary analysis, HOG-based automatic pedestrian detection algorithm has been applied to search for pedestrian from the raw video, and corresponding short video clips were generated. Combing these video clips and GPS-based map information, data was then manually extracted identifying the traffic controls (stoplights, stop signs, etc.), crosswalk, average speed of pedestrian, location, etc. There were around 62,000 pedestrian detections and of these there was about 3000 'potential conflict' (vehicle-pedestrian interaction) cases identified in the database. 'Potential conflict' is defined as if the direction and current speed of vehicle or pedestrian are not changed, they would cross at a point.

We have used machine learning techniques to analyze 1500 such 'potential conflict' videos to classify and understand the dynamic vehicle-pedestrian interactions. We have implemented visual tracker model [1–3] to recognize pedestrians and track them. Then, we have computed depth and lateral position of the pedestrians with respect to the vehicle. Next, we have employed the flexible mixture-of-parts method [4] to estimate human pose, and made some improvements to the basic algorithm. We have used this semantic behavior features data to inform a classification process, employing machine learning, to cluster behaviors into possible scenarios and provide an understanding of the significance of these new semantic features. We have also applied our method to TASI mannequin crash data to verify the accuracy of the algorithm. The results of this research can be used for developing autonomous driving rules, or for autonomous vehicle testing. Most part of the research has been presented in 45th IEEE Applied Imagery Pattern Recognition (AIPR) workshop (2016) in Washington D.C. and the paper is awaiting publication [5].

1.1 Objective and Motivation

Vehicle (driver)-pedestrian interactions such as: relative distances, trajectories, instantaneous velocities, and human pose changes all convey semantic information that can be used to interpret the behaviors of human subjects with respect to the vehicles autonomous or semi-autonomous driving system. The objective is to extract these data from existing naturalistic driving video through machine learning and tracking, modeling pedestrian-vehicle interactions in terms of crash avoidance and crossing/passing negotiation, and test the hypothesis through statistical analysis to explore these relationships.

According to National Highway and Traffic Safety Administration, in the US,the number of pedestrian fatalities in traffic crashes in the year 2015 was 5,376 [6]. If we are able to understand pedestrian behavior and predict their future movement, it is possible to reduce the number.

1.2 Literature Review

This research uses existing pedestrian detection method [7], modifies and improves the tracking and 3D tractography of our previous work [2], improves pose estimation from the existing literature [4], and combines it all into a machine learning classifier to detect 'potential conflict' situations. The literature is reviewed for each of these four cases.

1.2.1 Pedestrian Detection

A wide variety of techniques is used to detect pedestrians. Wavelet templates have been used as described in [8] to detect pedestrians in static images of cluttered scenes. The combination of local and global features via probabilistic top-down segmentation have been used to detect pedestrians in crowded scenes [9]. A template hierarchy and combined coarse-to-fine technique in shape and parameter space is used in [10] to detect pedestrians in moving vehicle. Pedestrian detection with unsupervised multi-stage feature learning is employed in [11]. In recent years, deep learning techniques have been used in a lot of research to detect pedestrians [12–15]. Deep network cascade, deformable template model, HOG based descriptor, etc. have been used for different real-time pedestrian detection systems [16–18]. Sixteen state-of-the-art pedestrian detectors have been evaluated across six different datasets in [19]. We have used pedestrian detection method using multimodal HOG for extracting pedestrian features as described in [7].

1.2.2 Pedestrian Tracking

A shape model for pedestrians and an effective variant of the condensation tracker is employed in [20] to track pedestrians from moving vehicle. Pedestrian activity can be understood and classified with motion history image, HOG and SVM in [21]. A combination of Kalman filter and mean shift tracking has been used to track pedes-

trian with night vision camera [22]. A prototype of automotive Pedestrian Protection Systems (PPS) has been implemented with a passive stereo vision configuration to have 3D vision sensing for pedestrian tracking [23]. Along with vision, LIDAR work with gaussian mixture model classifier and adaBoost classifier to detect and track pedestrian in [24]. Robust multi-person pedestrian tracking has been demonstrated in [25–27] using different techniques. For tracking in real-time, HOG and template matching approach have been employed in [28–30]. We have used adaptive structural local sparse appearance model and particle filter method to track pedestrians as outlined in [1, 3].

1.2.3 Pose Estimation

Flexible mixture model for encapsulating contextual co-occurrence relations between parts has been used for pose estimation in [4, 31]. Fast pose estimation methods have been developed using hash functions and iterative optimization in [32, 33]. Tracklet-based estimations have been described in [34] to have monocular 3D pose estimation. In contemporary research work, deep learning has been used extensively for pose estimation. Deep Neural Network (DNN), Deep Convolutional Network, and multi-source deep learning has been used to find pose estimation [35–38]. We have used articulated pedestrian pose estimation method using flexible mixture parts as described in [4, 31].

1.2.4 Combination for Classification

There has been a lot of research on autonomous and semi-autonomous driving recently. The prospect of autonomous driving in all situations is still a challenging problem. The long term challenges that have to be overcome in vehicle safety in autonomous driving, especially in urban environments, is described in [39]. In [40], challenges faced by an autonomous vehicle named ‘Boss’, equipped with GPS, lasers, radars and camera, is outlined.

Several crash avoidance system prototypes have been developed understand collision situations and test the designed system. Mannequins have been developed for evaluation of pre-collision systems in [41]. Collision avoidance model has been developed with predictive control with multi-constraints in [42]. Semi-autonomous multi-vehicle collision avoidance algorithm has experimented in an intersection testbed in [43]. Our research on this area is new and different compared to what others have done. We have used tractography and pose data to train a neural network to classify between 'potential conflict' and 'no potential conflict' situations, and also find the probability of collision for each second.

1.3 Our Contributions

This research explores challenges on autonomous driving scenarios and pedestrian behavioral feature extractions. Our primary contributions to this research are:

- Improving pedestrian tracking algorithm for moving camera for more accurate tractography.
- Developing new algorithm for calculating Focus of Expansion automatically for automating tractography.
- Predicting intermediate frame body pose of pedestrian if the pose is wrong or unrecognizable.
- Estimating stable and continuous pedestrian pose in frames using distance factor, Kalman filter and smoothing filter.
- Combining the tractography and pose data into a classifier achieving high accuracy of detecting potential conflict situations.

1.4 Organization

Our detailed research work is presented in the next three chapters and the dissertation concludes in the fifth chapter. In the next chapter, chapter 2, we discuss the theory and implementation process of the tractography. In chapter 3, we talk about the theory, modifications and improvements of pose estimation. In chapter 4, we present the results and statistically analyze the data.

2. TRACTOGRAPHY

To understand pedestrian semantic behavior, it is important to understand the dynamic relative positions of the vehicle and the pedestrian. Therefore, we need a trace of this data over time. The total process of the tractography data extraction is shown in the block diagram of Fig. 2.1. In this chapter, we discuss the techniques in details that have been used to determine pedestrian-vehicle spatial position. We complete the task in three steps as following:

1. Tracking the pedestrian.
2. Calculating Focus of Expansion.
3. Creating tractograph from the data obtained in first two steps.

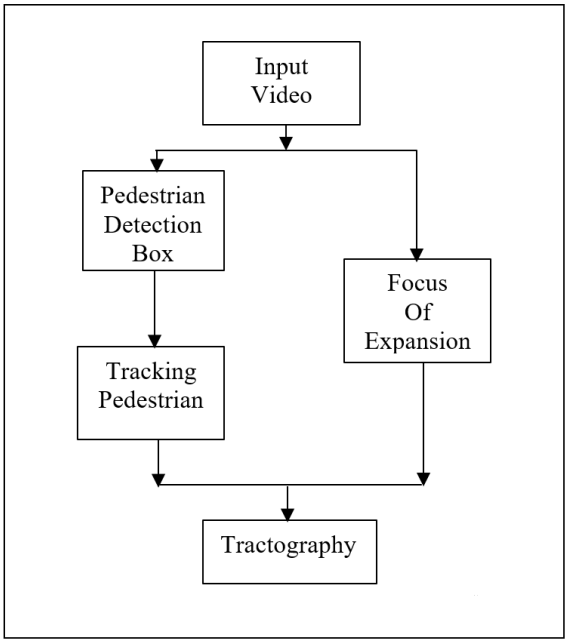

Fig. 2.1. Block diagram for tractography data extraction.

2.1 Tracking

Precise tractography data largely depends on the accuracy of the tracking. The tracking is important for two reasons:

1. The pedestrian must be tracked well in each frame to create an accurate graph.
2. The dimension of the tracking box around the pedestrian is later used for calculating depth and lateral position.

In this case we have used the adaptive structural local appearance model and particle filter methods described in [1–3]. From this base algorithm, we have done extensive experiments with these methods and modified several parameters so the code is best customized for pedestrians.

2.1.1 Pedestrian Detection

Pedestrians were extracted beforehand using (HOG-based) pattern recognition techniques [7] from the TASI 100 car naturalistic video dataset. After detection of a single pedestrian (verified manually and best pedestrian image frame chosen manually), these videos were organized into 5 seconds and 15 second videos centered in time on the detected pedestrian single frame. A database of manually identified features was made, and we used one of these features as the starting point for this research: potential conflict. Potential conflict is defined as if the direction and current speed of vehicle or pedestrian would cross at a point. Because we had no crashes, this implied that either the vehicle or the pedestrian changed speed or trajectory to avoid the collision. We have used these 15 second videos with potential conflict cases as our analysis starting point. Each video in the database has a log that contains some data analyst information, including position boxes of pedestrians and bicyclists in a frame and the reference frame number (out of the 15 seconds of frames). In our research, we used this log file for the starting frame and position box to further track the pedestrian in both directions (forward and back in time) from this center frame.

2.1.2 Particle Filter Method

The particle filter gives an approximation of the posterior distribution of a random variable that is related to a Markov chain as described in [3]. It gives a key tool for estimating the target in the next video frame without knowing the actual observation probability. The method consists of two major steps: prediction and update.

At the frame t, x_t describes the shape and location of the pedestrian. The observation of the pedestrian from the first frame to the frame $t1$ is denoted by $y_{1:t1} = \{y_1, y_2, ..., y_{t1}\}$. The filter proceeds two steps mentioned above with the following two probabilities:

$$p(x_t|y_{1:t-1}) = \int p(x_t|x_{t-1})p(x_{t-1}|y_{1:t-1})dx_{t-1} \tag{2.1}$$

$$p(x_t|y_{1:t}) = \frac{p(y_t|x_t)p(x_t|y_{1:t-1})}{p(y_t|y_{1:t-1}} \tag{2.2}$$

The maximal approximate posterior probability is used to find the optimal state for the frame t is as follows: $x_t^* = argmax(p(x|y_{1:t}))$. The posterior probability as described in equation 2.2 is estimated by using finite samples $S_t = \{x_t^1, x_t^2, ..., x_t^N\}$ with different weights $W = \{w_t^1, w_t^2, ..., w_t^N\}$ where N represents the number of samples. Using sequential importance distribution $\prod(x_t|y_{1:t}, x_{1:t-1})$, the samples are generated and weights are updated by:

$$w_t^i \propto w_{t-1}^i \frac{p(y_t|x_t^i)p(x_t^i|x_{t-1}^i)}{\prod(x_t|y_{1:t}, x_{1:t-1})} \tag{2.3}$$

In the case of $\prod(x_t|y_{1:t}, x_{1:t-1}) = p(x_t|x_{t1})$, the equation 2.3 has a rather simple form $w_t^i \propto w_{t1}^i p(y_t|x_t^i)$. To avoid degenerate increase of particle weights, in every step, samples are re-sampled to generate new sample set corresponding to their weights distribution. So, the weights in new sample set reflect the similarity between a target pedestrian candidate and target pedestrian template.

2.1.3 Implementation Process

Using the initial pedestrian detection described in section 2.1.2, we use the particle filter method to find pedestrian in all frames from the center frame [2]. The implementation process is given below:

1. The target template is obtained from the reference frame using the log information as shown in Fig. 2.2. Then we apply an affine transformation to make a customized template size.

Fig. 2.2. Target template from the reference frame.

2. Using particle filter method, target candidates or particles are generated. In our case the number of generated particle is 600. A Gaussian distribution is employed to model the state transition distribution. An example of 7 particle windows are shown in Fig. 2.3.

3. Region of interest (ROI) from the image is cropped by applying an affine transformation in Fig. 2.3 using the state information of the pedestrian as parameters

Fig. 2.3. Generated particles.

described in section 2.1.2. Then, the cropped image is normalized to have its dimension same as the dimensions of the target template.

Fig. 2.4. Cropped ROI with affine transform.

4. The similarity between the target candidates and the target template is calculated and the most similar one is found. Five target templates from Fig. 2.3 and the most similar with a circle is shown in Fig. 2.4.

5. The weights of particles are updated based on the computed similarity results.

The process is repeated in both temporal directions from the reference frame. The tracked box positions for all frames are stored to be used later for the tractography plot.

2.2 Focus of Expansion

The Focus of Expansion (FoE) is an important parameter to compute the relative distance between the pedestrian and vehicle, and the height parameter of the FoE strongly effects the depth calculation in the tractography as showed in the error analysis in [2]. We have improved the automation of the FoE calculation. We calculate this automated focus of expansion using the following process:

1. Take 30 frames (a 1-second clip) of a sequence where the vehicle is moving in the videos, then average them, forming a single image. This produces a smear of the video, centered at the FoE. This is shown in the left column of Fig. 2.5.

2. Apply a Hough transform to find lines in the image which will converge to the FoE, along the smeared video. Some lines are then eliminated based on orientation angle, as the FoE is expected only in the center of the image, and must pass through a center ROI. These remaining lines are extended in both directions and are shown in the center column of Fig. 2.5.

3. Calculate the FoE from the intersection of lines using the center of mass of the crossing points of the lines as the expected FoE, as shown in the right column of Fig. 2.5 (small red ‘x’ in FoE center).

Fig. 2.5. Focus of Expansion. Averaged image (left column), Extended hough lines (center column), FoE detection (right column).

We have applied several conditions and imposed restrictions to find out the accurate FoE. For more accuracy, we have calculated FoE for each second and selected the best one (highest number of lines, with low variance for the crossing points) for the total video.

We have used automatically generated FoE data to compare with the carefully clicked FoE data by humans. We have stated above that the height parameter (y) of FoE strongly affects the depth calculation. So, it is very important to have an accurate height measurement from automatic FoE algorithm. The automatically detected FoE height have a correlation of 0.98 with the human clicked FoE height. However, the width parameter has a correlation of 0.70. The result is not as good for the width parameter (x) mainly when the car changes its direction, the FoE width moves in the same direction. From the tractography, this performance is adequate, as it does not affect the error as strongly. The results are plotted in figures 2.6 and 2.7.

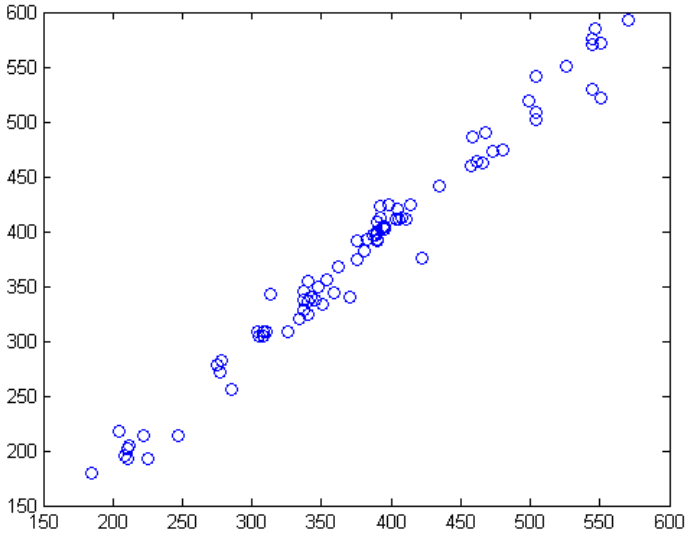

Fig. 2.6. Comparing height parameter of automatic vs. human clicked FoE.

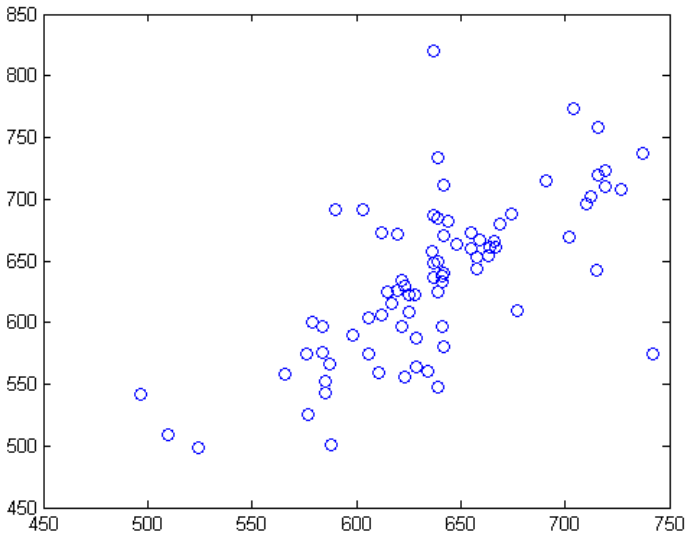

Fig. 2.7. Comparing width parameter of automatic vs. human clicked FoE.

2.3 Tractography Data Extraction

2.3.1 Depth and Lateral Position Calculation

We have developed a prototype of calculating distance from camera to pedestrian (depth) and lateral position using computer vision techniques combined with 2D to

3D projection geometries. The geometric relationship between a pedestrian and the pedestrian projection of the image plane is shown in Fig. 2.8 as described in [44, 45].

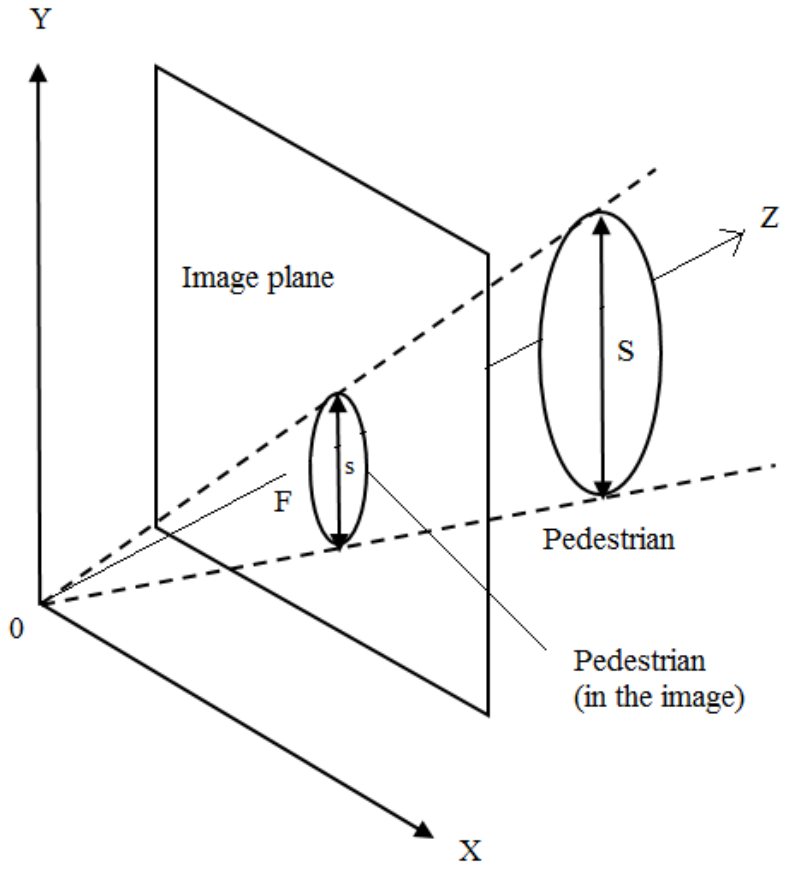

Fig. 2.8. Pedestrian projected in the image plane.

In Fig. 2.8, X, Y, Z denotes euclidean coordinates, F denotes focal length, S denotes size of the pedestrian and s denotes size of the pedestrian in the image plane. We can get the following equation from the geometric relationship of Fig. 2.8.

$$\frac{1}{s} = \frac{Z}{S * F} \tag{2.4}$$

The distance of the pedestrian from camera was calculated first to determine the lateral position as described in [2] and [46]. To do so, the geometry of the pedestrian is shown Fig. 2.9.

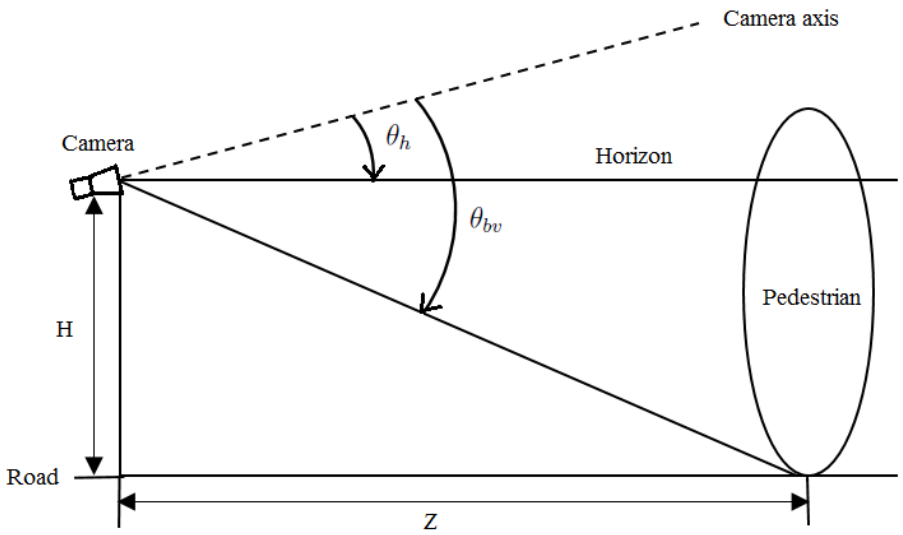

Fig. 2.9. Geometry of pedestrian (side view).

In the Fig. 2.9, Z represents the distance of pedestrian from camera (depth), H represents height of the camera, θ_h represents the vertical angle of the horizon with the camera axis, and θ_{bv} represents the vertical angle of the pedestrian's bottom with camera axis. If the horizon and the pedestrian's bottom edge are defined in the image, the angles of θ_h and θ_{bh} can be determined. In our case, the horizon was specified with FoE and the pedestrian's bottom edge was found in tracking part, both described above. The equation 2.5 is used to calculate distance Z.

$$Z = \frac{H}{tan(\theta_{bv} - \theta_h)}) \tag{2.5}$$

To determine the lateral position, we need a top view geometry of the pedestrian. This is shown in Fig. 2.10. In the figure, X_c and Z_c signify pedestrian position in the coordinate system with respect to the camera, X_v and Z_v signify pedestrian position in the coordinate system with respect to the camera, θ_c signifies the angle between vehicle moving direction and camera axis, and θ_{bh} signifies the horizontal angle of the pedestrian in the camera coordinate system. The vehicle moving direction was found

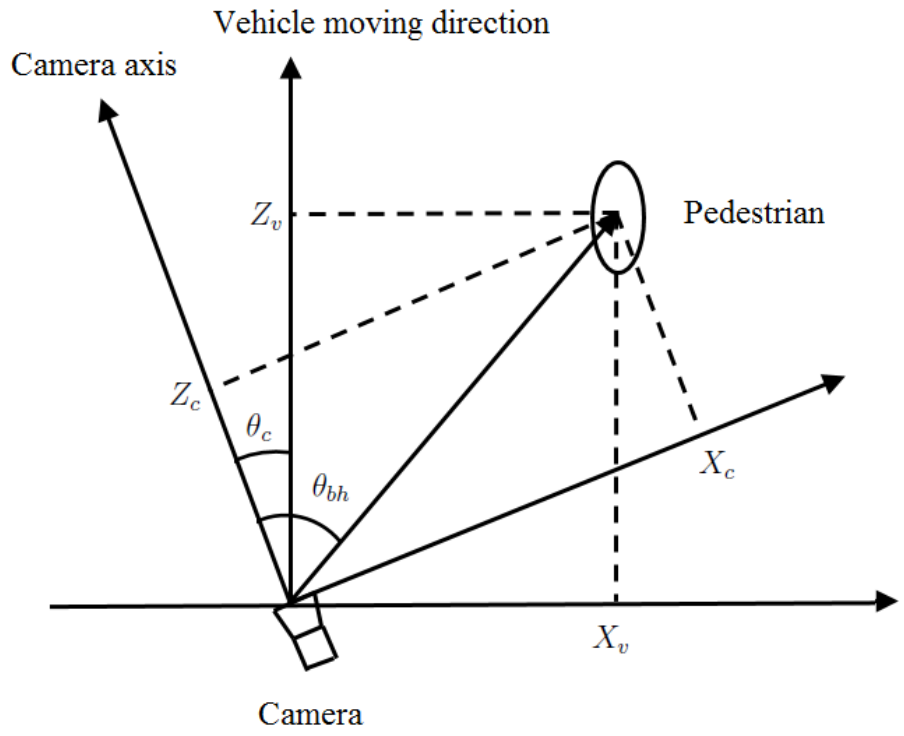

Fig. 2.10. Geometry of pedestrian (top view).

from the FoE, and the θ_{bh} was computed from the tracking rectangle. The rotation of the coordinate systems is shown in equation 2.6 and 2.7, which are used to compute lateral position X_v.

$$X_c = Z_c tan\theta_{bh} \tag{2.6}$$

$$\begin{bmatrix} X_v \\ Z_v \end{bmatrix} = \begin{bmatrix} cos\theta_c & -sin\theta_c \\ sin\theta_c & cos\theta_c \end{bmatrix} \begin{bmatrix} X_c \\ Z_c \end{bmatrix} \tag{2.7}$$

2.3.2 Tractography Results

Understanding of the vehicle-pedestrian dynamic position is very important for understanding of vehicle-pedestrian semantic behavior. Therefore, we need to plot the

relative distance of pedestrian from vehicle over time. The time series of the tracks can be collected together and visualized as a tractograph as shown in Fig. 2.11.

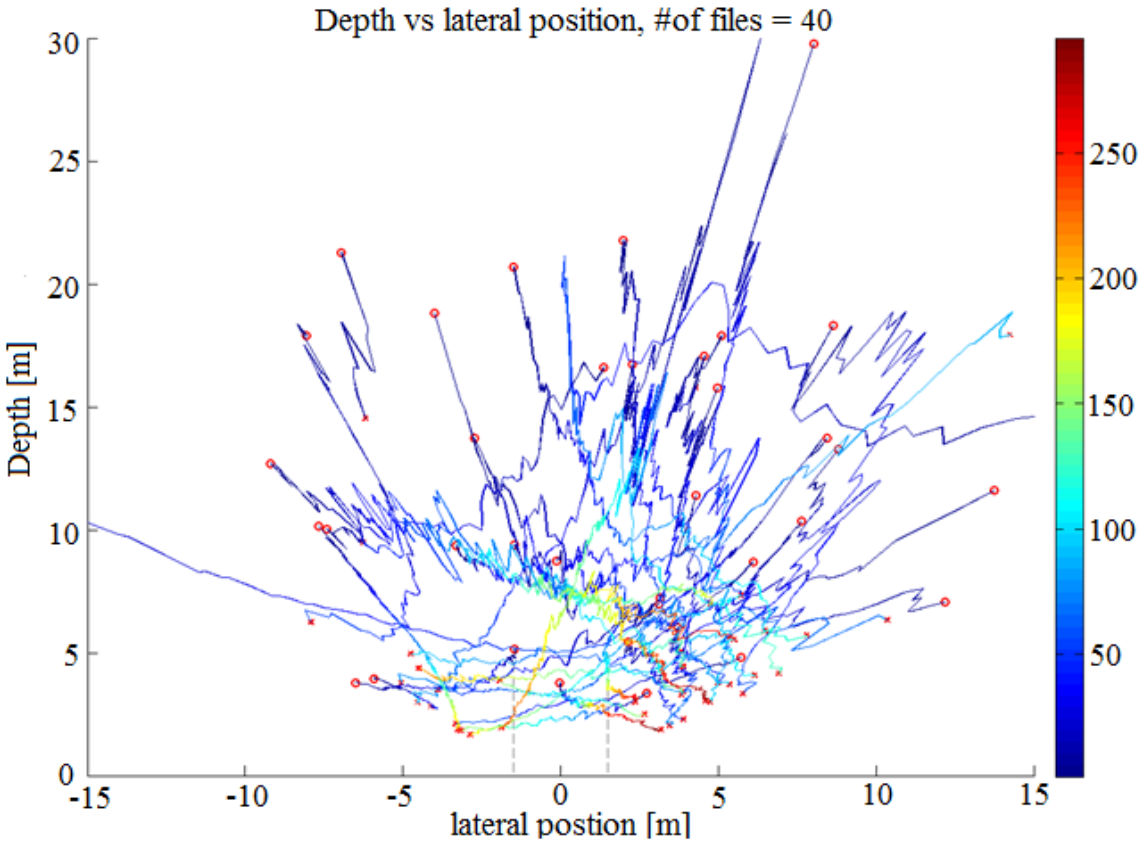

Fig. 2.11. Tractograph of 40 videos.

In this figure, the instantaneous positions of 40 randomly chosen pedestrian videos have been overlaid into a single plot. The graph is relative position of the pedestrian with respect to the vehicle position, where the vehicle is centered anchored between the dashed lines at the center-bottom of the graph. The start point (first appearance of the pedestrian in time) is denoted by red "o" and end point (disappearance point) is denoted by red "x". The direction of the movement with time is denoted by the color of the trace as shown in the sidebar of the figure. The sidebar scale represents frame number at 30 frames per second. So, from the color of a trace at the end point we can understand how much time that trace denotes. The features from the relative

positions and dynamic behaviors developed from this tractography can also be used to inform the semantic human-vehicle interaction. Later in the data analysis, large variations may be smoothed with filtering or outliers eliminated to produce smoother tracks. Also, we know from our previous work [2], that the position data accuracy reduces with distance from the car, so the best region of interest for accurate data will be in a half circle in front of the car. Typical scenarios can then be gleaned from this data that are useful for autonomous driving control or for testing such vehicle systems.

From the database of our TASI 110-car study, human generated vehicle-pedestrian motion has tagged this pedestrian motion into four different broad categories:

1. Pedestrian crossing from right to left (of the vehicle).
2. Pedestrian crossing left to right.
3. Pedestrian walking towards the vehicle.
4. Pedestrian walking in the same direction as the vehicle.

These four scenarios are shown in figures 2.12 to 2.15, collecting 15 cases of each scenario.

Tractography can give us significant insight of vehicle-pedestrian negotiation. For example, when pedestrian walking in the opposite direction of the vehicle the relative distance decreases very fast. We can see that for some cases in Fig. 2.14 relative distance (depth) has decreased a lot in just a small period of time which is visualized by a little change in color. This we can glean from the color change in the trace. This understating can be an important factor for calculating risk factor, time to collision and other parameters which are essential for autonomous driving.

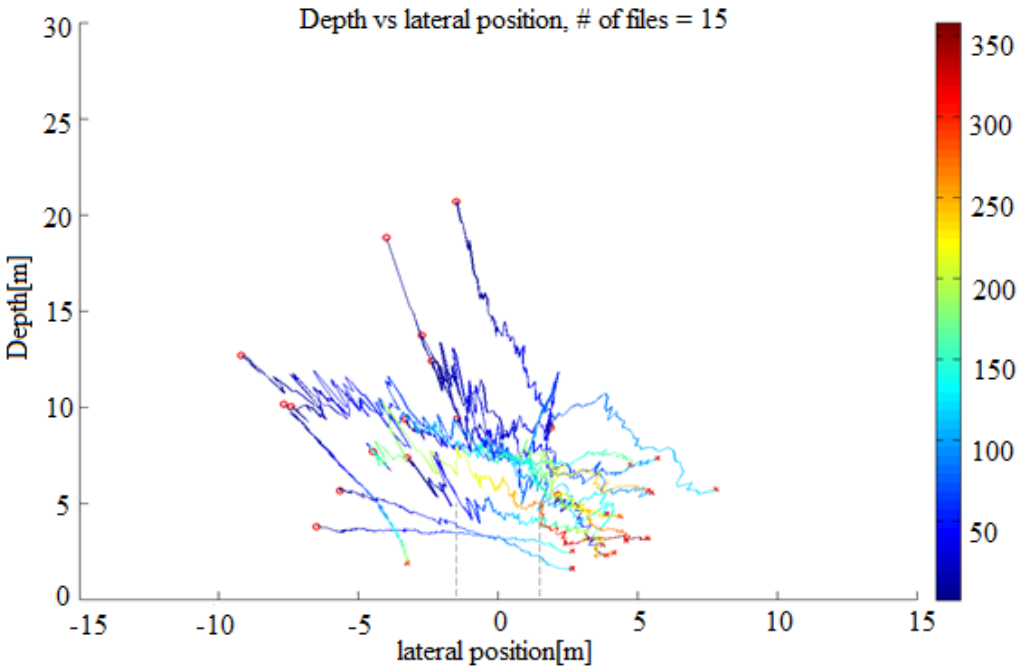

Fig. 2.12. Pedestrians crossing from left to right.

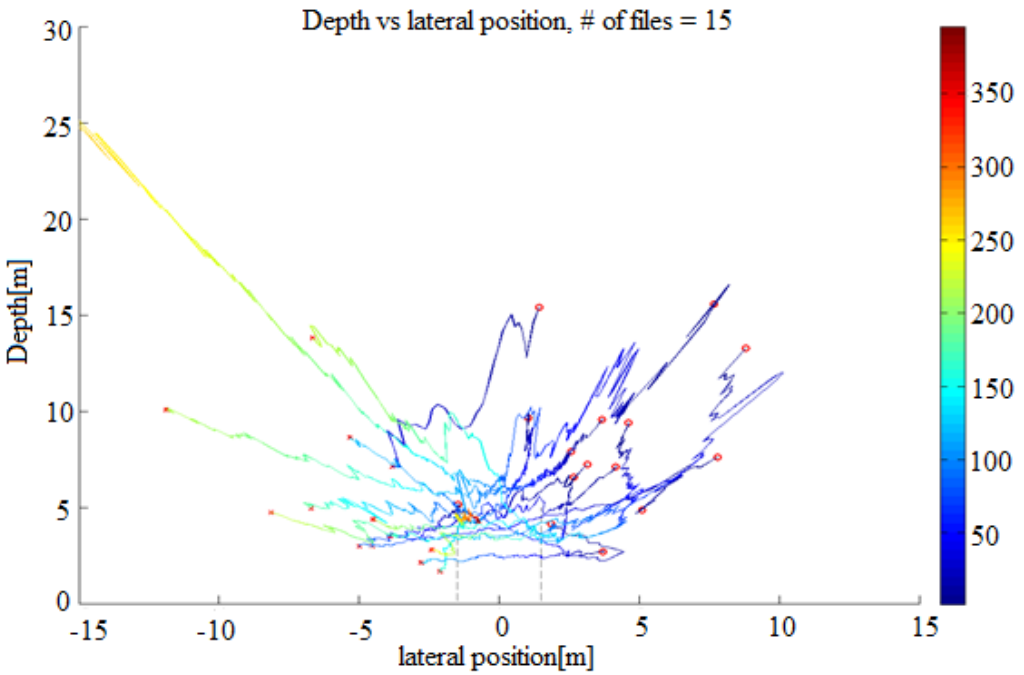

Fig. 2.13. Pedestrian crossing from right to left.

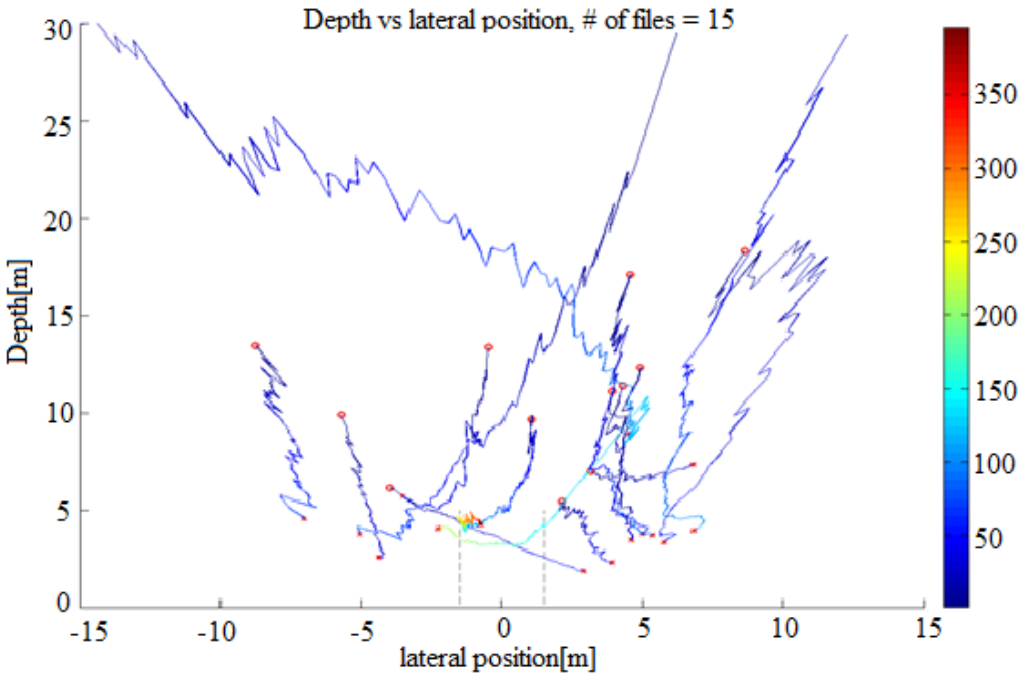

Fig. 2.14. Pedestrians walking towards the vehicle.

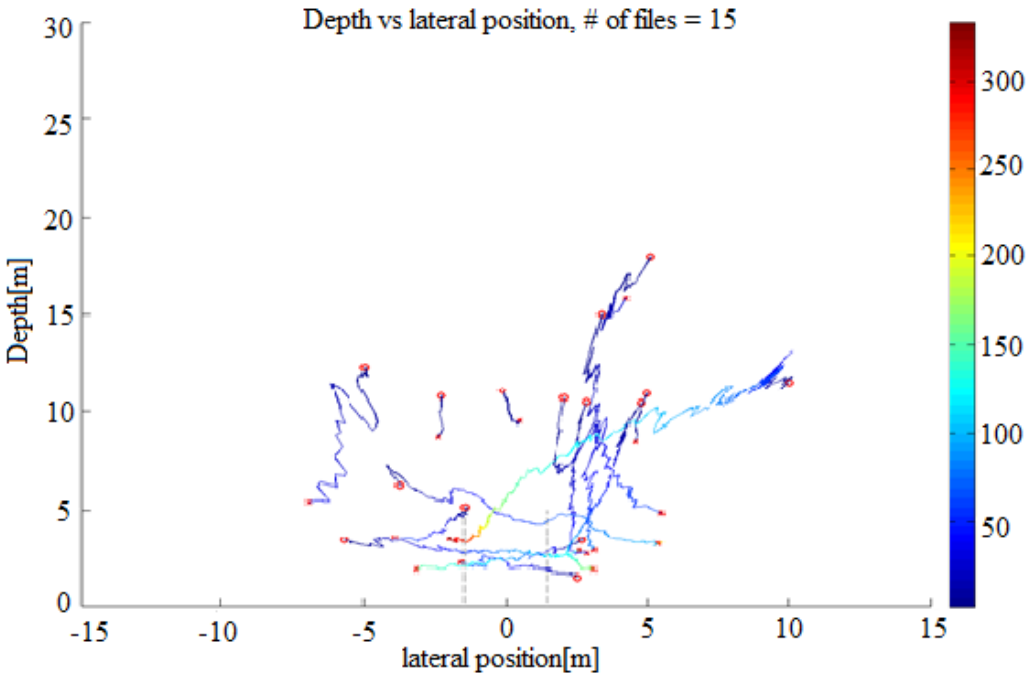

Fig. 2.15. Pedestrians walking with the vehicle.

2.4 Conclusion

In this chapter, we have discussed the theory and implementation process of the tracking algorithm based on adaptive structural local appearance model and particle filter methods. Then, we detailed how we used a novel method to calculate FoE automatically. Finally, we have created tractograph using tracking and FoE data and interpreted the graphs. In the next chapter, we will dive into the details of our modified pose estimation algorithm.

3. POSE ESTIMATION

Pose estimation is very important for understanding pedestrian semantic behavior. We have used an articulated flexible mixture model for human pose estimation in static images based on a representation of part models described in [4]. The total process of pose estimation is shown in the block diagram in Fig. 3.1.

3.1 Video Frame Processing

Frames of a video are cropped using the pedestrian position information from the tracking. The cropped images are little larger than the box size. In our case, we will provide these cropped images from tracking as input rather than the whole image to reduce computational time.

3.2 Image Preprocessing

For image preprocessing, initially we used histogram equalization and adaptive histogram equalization. Though adaptive histogram equalization performed better than histogram equalization, we experimented with different sharpening filters for more accuracy. Sharpening filter with an unsharp masking performed better than adaptive histogram equalization. However, adaptive histogram equalization along with the sharpening filter performed almost as good as the sharpening filter alone. So, we used the sharpening filter alone and also both sharpening filter and adaptive histogram equalization interchangeably to boost the high frequency components of the images. This is a very important finding from our current research that for naturalistic videos sharpening alone or combination of sharpening and adaptive histogram equalization perform better than solo histogram equalization or adaptive histogram

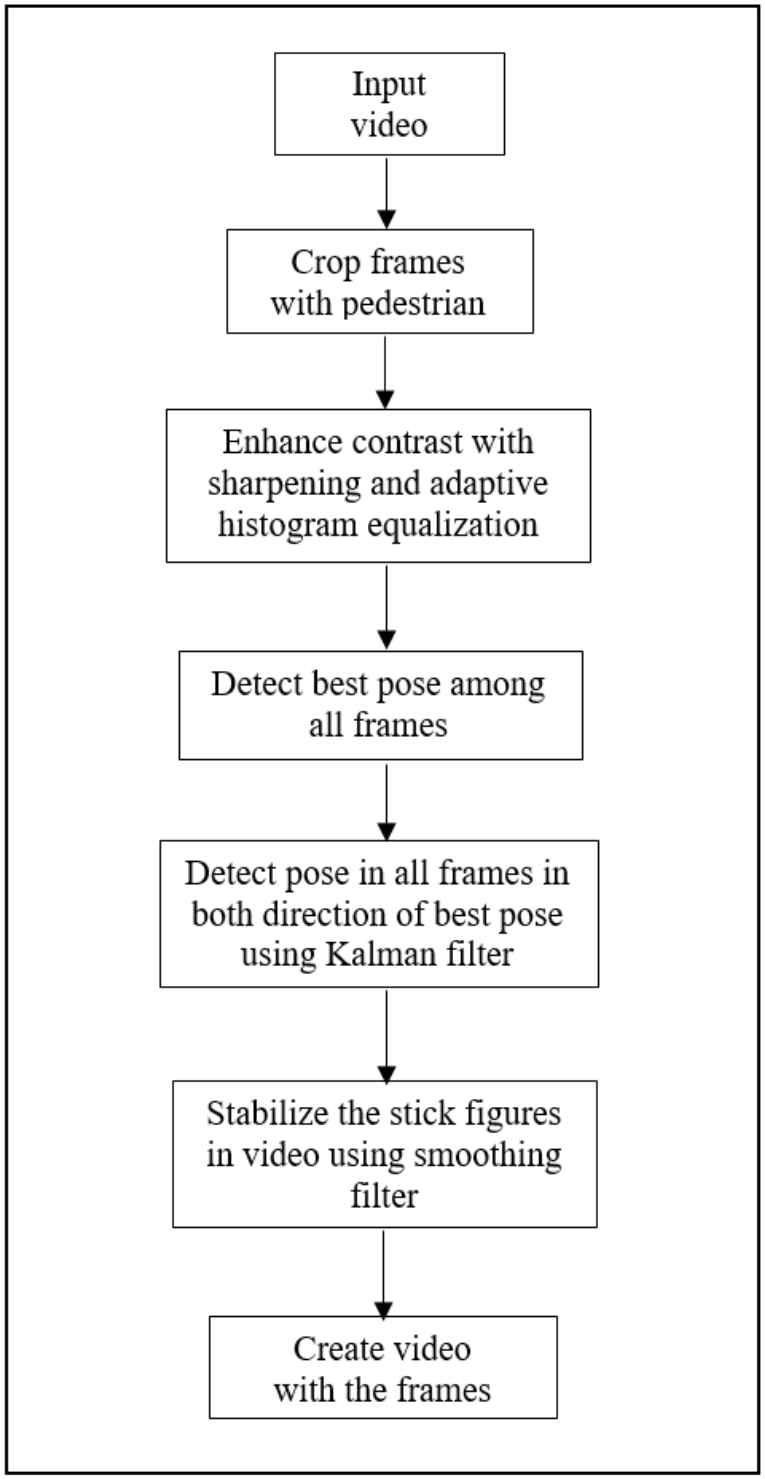

Fig. 3.1. Block diagram for pose estimation.

equalization. The original paper [4] used very good high-contrast, high resolution human figure pictures. Our data has varying resolutions and contrast, due to the natural light variation across the day, and the distance to the pedestrian.

3.3 Articulated Pose Estimation with Flexible Mixture of Parts

The pose estimation method that we used does not use articulated limb parts, but rather captures how the templates of each part orient with each other. A general, flexible mixture model is used for capturing co-occurrence relations between segments. Then, standard spring models are augmented that encode spatial relations. It has been shown in [4] that such relations can capture notions of underlying local structure. The model can be effectively optimized with dynamic programming when co-occurrence and spatial relations are tree-structured.

A feature pyramid is created for each image considering all limb parts of the human. Corresponding confidence scores are also calculated for each limb part. Then a dynamic programming algorithm is implemented to select the best combination of the parts and a corresponding confidence score of the whole human pose is also calculated. The pedestrian is modeled with 14 part articulated parts as shown in Fig. 3.2.

Let, the image is represented with I. In the image, $p_i = (x, y)$ denotes the pixel location of part i and t_i denotes the mixture component of part i. In order to score of a configuration of parts, a compatibility function for part types is defined in equation 3.1 which considers sum of local and pairwise score [4].

$$S(t) = \sum_{i \in V} b_i^{t_i} + \sum_{ij \in E} b_{ij}^{t_i, t_j} \tag{3.1}$$

In above equation, particular co-occurrence of part types is favored by the pairwise parameter $b_{ij}^{t_i,t_j}$ while particular type assignment for part i is favored by parameter $b_i^{t_i}$. The full score corresponds to a configuration of part types and positions is calculated with equation 3.2 where $\phi(I, p_i)$ is a feature vector. The pedestrian parts

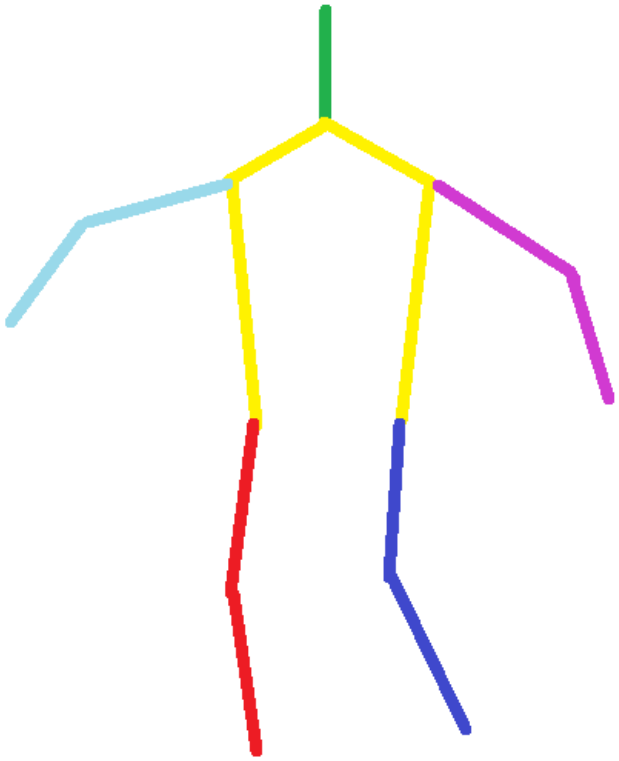

Fig. 3.2. Pedestrian body pose model.

are modeled with $G = (V, E)$, a (tree-structured) K-node relational graph, to learn which collections of parts are rigid to find corresponding pose estimation.

$$S(I, p, t) = S(t) + \sum_{i \in V} w_i^{t_i} \cdot \phi(I, p_i) + \sum_{ij \in E} w_{ij}^{t_i, t_j} \cdot \psi(p_i - p_j) \tag{3.2}$$

Using dynamic programming with tree-structured $G = (V, E)$ graph, the local score of part i and for every j, the best scoring position and location of part i are computed with equations 3.3 and 3.4 respectively.

$$score_i(t_i, p_i) = b_i^{t_i} + w_{t_i}^i \cdot \phi(I, p_i) + \sum_{k \in kids(i)} m_k(t_i, p_i) \tag{3.3}$$

$$m_i(t_j, p_j) = \max_{t_i} b_{ij}^{t_i, t_j} + \max_{t_i} score(t_i, p_i) + w^{t_i, t_j} \cdot \psi(p_i - p_j) \tag{3.4}$$

3.4 Temporal Movement Factor

From the original algorithm, we have improved the result by using temporal information. At this step, we choose the single best (highest confidence score) frame of the pose estimation in the video sequence. Then, the algorithm is modified to extend in both temporal directions to estimate the new adapted human pose. For this purpose, we track the confidence scores and we also consider relative position of the human limb parts in the frame compared to previous three frames, as human movement should be small from frame to frame. The new algorithm then selects the best pose using confidence score and relative distances. It is shown in Fig. 3.3.

Fig. 3.3. The top row stick figures with original algorithm [4]. The bottom row stick figures with the new improved algorithm with pre-processed images and temporal information.

3.5 Kalman Filter

To further improve the result, we have used Kalman filter to estimate the future location of the different body parts that has been used to create the estimated stick figures. It improved the result significantly, especially with the frames where there was missing pedestrian pose data. We could predict the stick figure joint locations of the non-detected frame with the Kalman filter. The filter also did a very good job at reducing the weight of the wrong detections among frames.

3.6 Smoothing Filter

After the use of Kalman filter, a moving average smoothing filter was to smooth the changes of positions of the stick figures. The final output is the stable and smooth stick figures of the pedestrian poses.

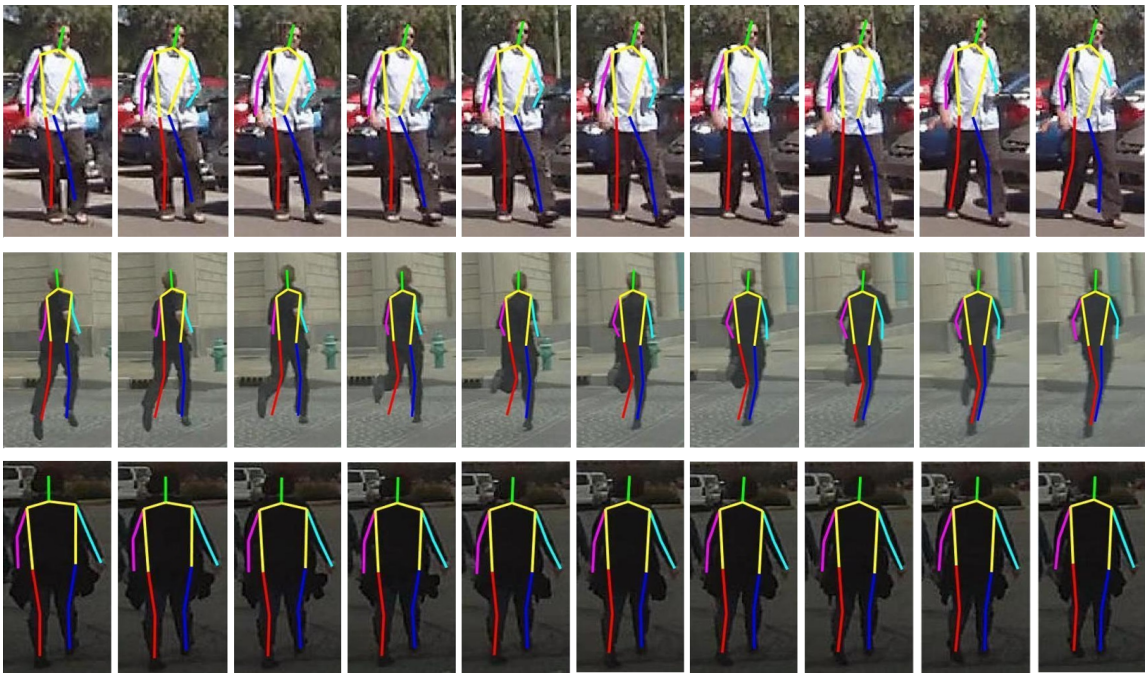

Fig. 3.4. Pedestrian pose estimation.

3.7 Final Results

Using the above described method, we have been able to reduce frame to frame pixel offset by 86% compared with the previous single frame model. Ten consecutive output frames for pedestrian pose estimation for three videos are shown in Fig. 3.4.

3.8 Conclusion

In this chapter, we have discussed the theory and implementation of the pose estimation step by step. It includes how we did the image preprocessing, how we did different filtering, and how the original algorithm was modified to suit our purpose. In the next chapter, we will analyze the data that we computed in tractography and pose estimation.

4. CLASSIFICATION AND STATISTICAL DATA ANALYSIS

We have analyzed the data that we obtained from tractography and pose estimation in this section. We have used neural network to train a classifier using tractography and pose data to identify and classify 'potential conflict' situations. The reasons for choosing neural network includes [47]:

1. An ability to learn how to perform a task based on the given training data.
2. Neural network can create its own representation of the information when receiving during the training time.
3. Neural network is fault tolerant via redundant information coding.

We haven't used deep learning because the size of the training dataset is not large enough. We have used feature selection to check if we can remove some redundant features from training set.

We have used 34 examples of the 5-second video sequences that were human-labeled with 'potential conflict' signifying that at some point in the video, the path of the pedestrian and the path of the automobile would cross (without avoidance behavior). Since none of the vehicles in our study encountered a true crash, we were using the potential conflict as a training set for the statistical analysis. We also obtained 34 vehicle-pedestrian 5 seconds videos that were labeled 'no potential conflict'. For example, a pedestrian on a sidewalk parallel to the motion of the vehicle is considered 'no potential conflict'. This data for 68 videos was used for feature selection and neural network training.

4.1 Feature Selection

Feature Selection can be a useful tool in the neural network in reducing the number of features to train the network. But for our case, the tractography data is a time series of data. So, if the series of data is used to train a neural network we can find out which data points are most important. Again, we can check the pose data if any articulated part is more prominent in determining if a situation is a potential conflict or no potential conflict. For 5 seconds videos we have 150 frames. We have used 150 tractography data points as 150 features. Here, Principal Component Analysis (PCA) is done on the data point of (x,y) coordinates to make it a single feature. To get the general idea of how each feature can separate the groups we have the Cumulative Distribution Function (CDF) vs P value graph for the 150 features of tractography as shown in Fig. 4.1.

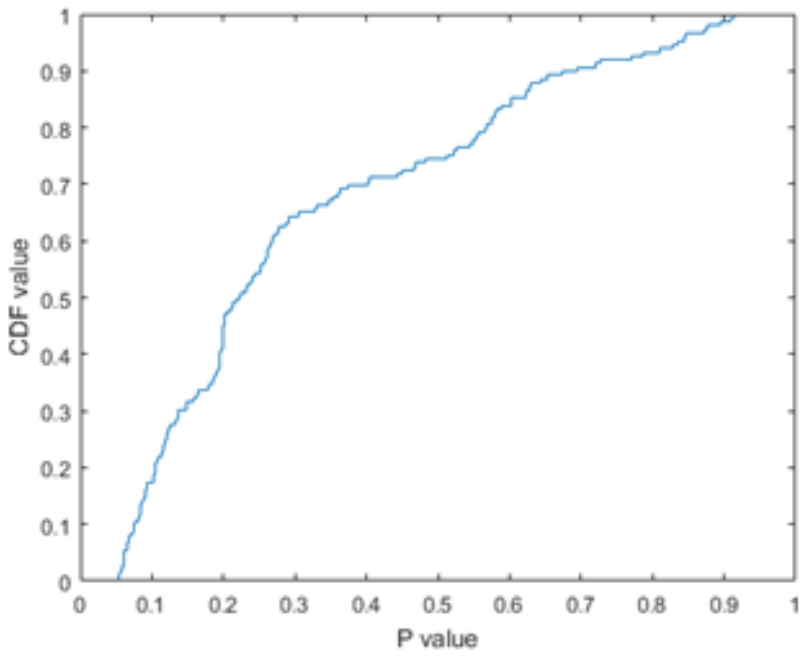

Fig. 4.1. CDF vs P value for tractography features.

As the CDF is zero for P values near zero, it is evident that no specific feature or very few features can successfully classify between the groups. Though some features are more dominant and some can be eliminated to reduce the feature dimension, the reduction is not significant. The result is actually consistent with our experimental result because as the features are time series of data, no single or few data points in the time series can successfully separate the groups.

We also used the PCA for pose estimation to have one feature for each articulated body part. To find the dominant articulated part or parts from pose estimation data which can classify between the groups we produced the graph shown in Fig. 4.2.

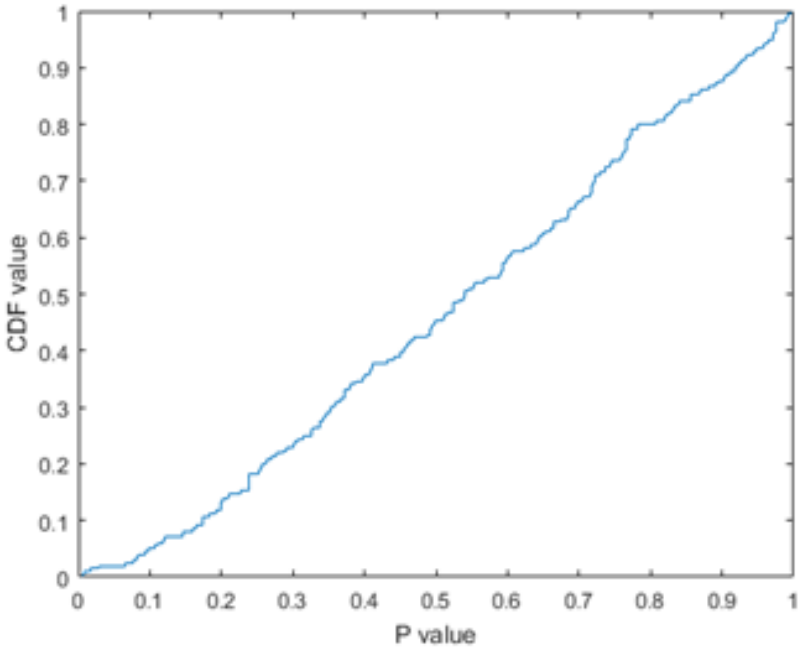

Fig. 4.2. CDF vs P value for articulated parts.

From the graph, we can understand that no single articulated part movement can successfully classify the groups. But, all parts comprehensively can be used to separate group for the case of pose estimation.

4.2 Neural Network Training

First the Principle Component Analysis (PCA) was done on the input data (tractography and pose) similarly as it was done for feature selection, and then a 2-layer neural network was trained with this input. We randomly sampled the input PCA data into 70% training, 15% validation, and 15% test. The training of this network achieved 91.2% true label accuracy, and 8.8% false label accuracy. This can be seen in the blue square of the All Confusion Matrix in Fig. 4.3.

Fig. 4.3. Confusion matrix for trained neural network.

It is common in neural net research that random sampling of test and training data and different initial conditions will affect the confusion matrices on 68 data size. In our case, as can be seen in Fig. 4.3, the result for test confusion matrix is not so good. We got better results for test confusion matrix for other trainings when the initial random test samples were different but we have used it because it had the highest true positive for all confusion matrix.

4.3 Danger Assessment

Since any real-time system will not have the advantage of the 'future' time of the whole interaction between the pedestrian and the vehicle, the data must be labeled over time, so this task is to run short segments of time through the trained network to identify the key features that indicate potential conflict outcomes. As from the tractography data we can extrapolate that there is a chance of collision if the path of the vehicle and pedestrian crosses, it can be used for used for continuous danger assessment over time.

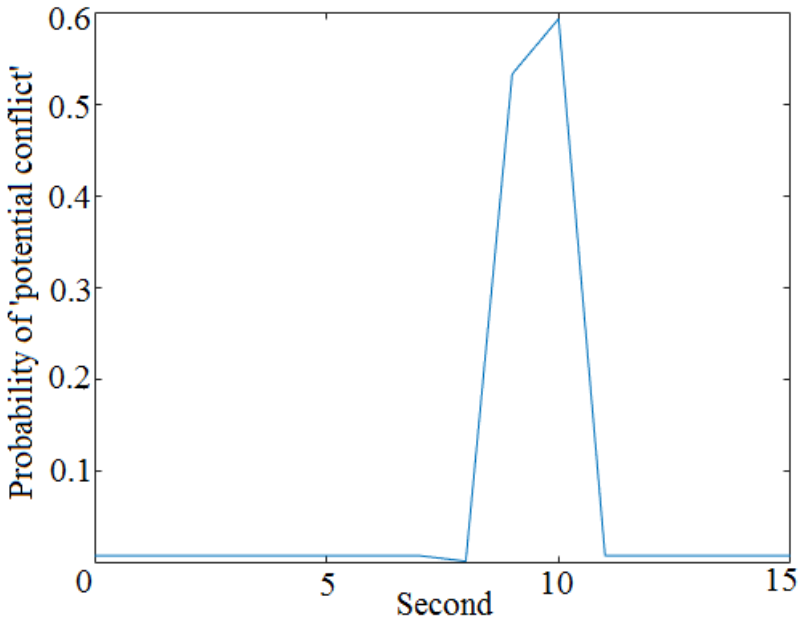

Fig. 4.4. Danger assessment in a 15 seconds video.

The neural network that we have trained to classify between potential conflict and no potential conflict situations was used to assess the risk of collision over time for the 15 seconds potential conflict videos, our main database for this project. Fig. 4.4 shows the probability of 'potential conflict' over 15 seconds of a video.

From Fig. 4.4, we can see that there is a chance of 'potential conflict' above the threshold of 0.5 value around 9-11 seconds. The accuracy of the graph can be verified manually by watching the video if really there is a 'potential conflict' situation in the video in the same time as the spike of the graph.

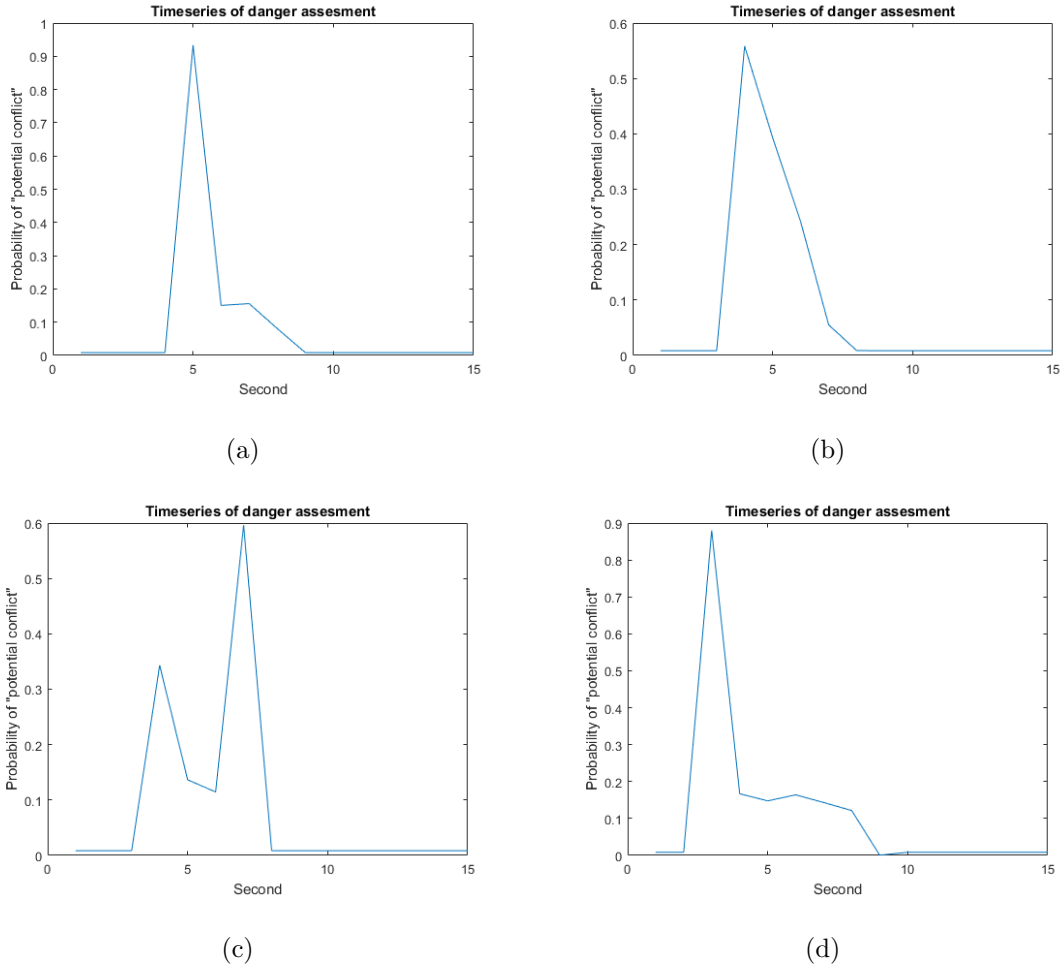

Fig. 4.5. Danger assessment of more potential conflict videos.

We have checked that for 90% cases the trained network can produce a spike when there is visibly a 'potential conflict' situation. Some more examples of danger assessment in potential conflict videos are shown in Fig. 4.5.

4.4 Mannequin Crash Data Analysis

In this section, we test our trained network with the mannequin crash data. Two frames of a standard crash video are shown in Fig. 4.6 to understand the set up.

The crash video was captured in a TASI test site using TASI designed mannequins which can be restored easily after crashing with a car [41]. We have used these crash video to do the tracking of the mannequins, find FoE of the camera and eventually tractography. We also did the pose estimation for the mannequins. Then, we used these data to find the probability of collision using the neural net which we have trained before. The danger assessments of some mannequin crash videos are shown in Fig. 4.7.

In the danger assessment graphs, we can clearly see that there is a high probability of collision and the mannequins did collude with the car. In two out of four videos, the probability of collision almost reaches '1.0' probability which signifies imminent collision possibility.

For 70% of the crash videos, we get a spike which indicates the menacing probability of collision. But in a numeric sense, it is lower than what we got in the previous section for 'potential conflict' videos. There are several reasons for that. The primary two reasons are:

1. When the pedestrian gets close to the car, the feet of the mannequins gets occluded by the front part of the car. The occlusion in turn makes the size of the mannequin smaller, resulting in apparent bigger distance from the car in the tractography.

2. The occlusion of the feet in the tracking part contributes to the deterioration of pose estimation as we use the tracking box to find pose estimation.

For the above mentioned reasons, the pose estimation results for mannequins are not as much accurate as pedestrians. In future, we will work to improve that.

Fig. 4.6. Two frames from a crash video.

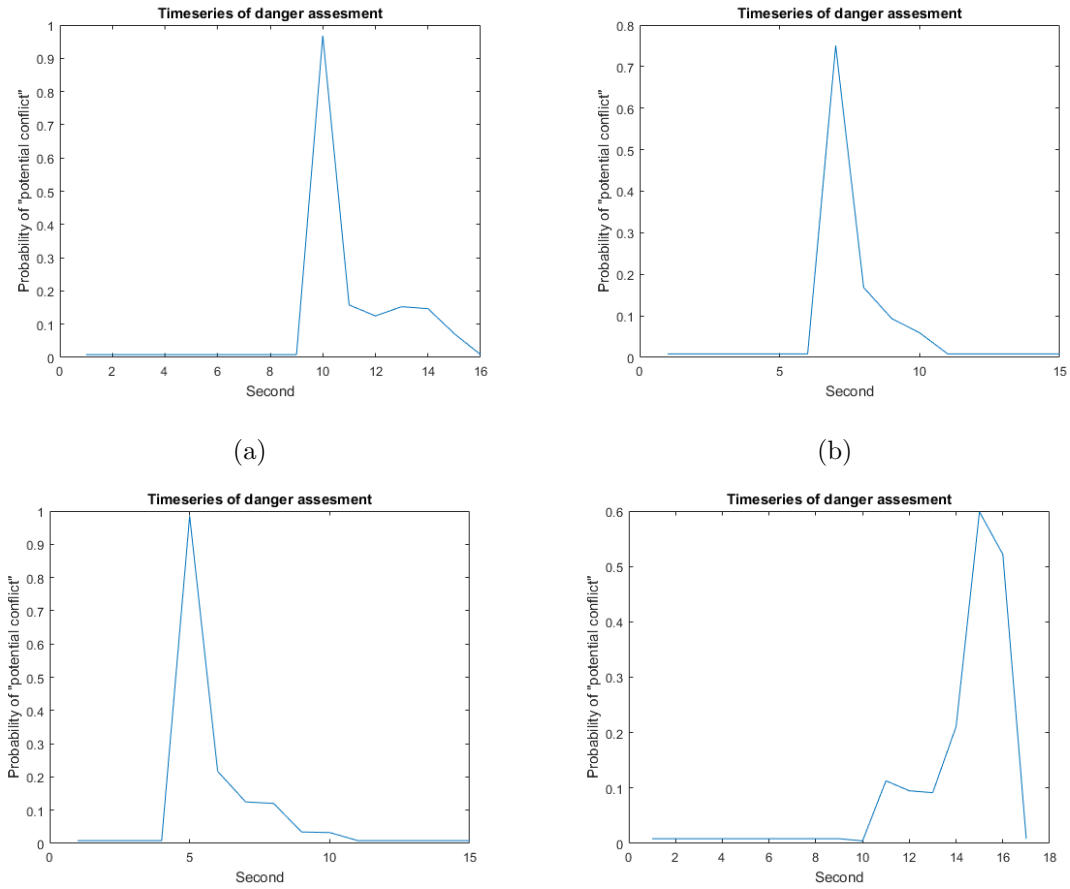

Fig. 4.7. Danger assessment of crash videos.

4.5 Conclusion

In this chapter, we have statistically analyzed the data obtained from chapter 2 and chapter 3. We have used feature selection and neural network to examine the data. We have also trained a network to access the probability of collision and tested the network with mannequin crash videos. In the next chapter, we will conclude the dissertation and state we plan to do in future to improve the research.

5. CONCLUSION

The long-term goal of the research is to enable advances in computer vision, robotics, vehicle safety, and consumer electronics by capturing human semantic information from naturalistic driving movies. This study will inform the future test scenarios and provide more advanced concepts for autonomous driving systems about the probable vehicle-pedestrian interactions.

As pedestrian-vehicle interaction is better understood, systems can be created to reduce confusion and wrong decisions from the two parties, improve traffic efficiencies, and prevent injuries or fatalities. In addition, the false alarms (false positives) from these autonomous or semi-autonomous driving systems can be reduced, if normal pedestrian behavior is understood.

5.1 Summary

In this research, we have improved the tracking algorithm to be better suited for pedestrians and obtained accurate tractography for more than 82% of the videos. We have developed a new algorithm to calculate Focus of Expansion automatically with a height parameter correlation of 0.98 with the carefully manually clicked data. We have been able to predict the future movement of the pedestrian using Kalman filter and temporal movement factor reducing 86% of frame to frame pixel offset. Even if the pedestrian is not detected or falsely detected, we can estimate pedestrian position from the previous frames. So, we can get continuous pedestrian pose for all frames of the video. Using all these information, we have successfully classified 90% pedestrian and 70% mannequin potential conflict cases.

5.2 Future Works

Even though we have achieved good results, there is room for improvements in our research. In future, we will work on increasing the percent of videos with accurate tracking. We would also like to work on improving the width parameter of the automatically detected FoE. For pose estimation, when there is occlusion, especially self-occlusion of pedestrian hand by the torso, the pose estimation doesn't work very well in that scenario. We will work to enhance the pose estimation accuracy in occluded scenes. To improve performance in occluded scenes, we should use only the top of the human to approximate overall size of the human and distance between car and pedestrian more precisely. Our major area of improvement in future would be an upgrade of neural net performance for crash videos. We would also like to use state-of-the-art deep learning techniques to understand vehicle-pedestrian semantic behavior.

REFERENCES

REFERENCES

[1] X. Jia, H. Lu, and M. H. Yang, "Visual tracking via adaptive structural local sparse appearance model," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 1822–1829.

[2] C. Liu, R. Fujishiro, L. Christopher, and J. Zheng, "Vehicle-Bicyclist Dynamic Position Extracted From Naturalistic Driving Videos," *IEEE Transactions on Intelligent Transportation Systems*, vol. PP, no. 99, pp. 1–9, 2016.

[3] C. Bao, Y. Wu, H. Ling, and H. Ji, "Real time robust l1 tracker using accelerated proximal gradient approach," in *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2012, pp. 1830–1837.

[4] Y. Yang and D. Ramanan, "Articulated Human Detection with Flexible Mixtures of Parts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 12, pp. 2878–2890, 2013.

[5] R. Mueid, L. Christopher, and R. Tian, "Vehicle-pedestrian dynamic interaction through tractography of relative movements and articulated pedestrian pose estimation," in *2016 IEEE Applied Imagery Pattern Recognition Workshop (AIPR)*. IEEE, 2016 (Estimated publication date: 05/2017), pp. 1–6.

[6] N. Highway and T. S. Administration, "Traffic safety facts: 2015," 2017, date accessed: 03-29-2017. [Online]. Available: https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/812375

[7] K. Yang, E. Y. Du, E. J. Delp, P. Jiang, F. Jiang, Y. Chen, R. Sherony, and H. Takahashi, "An extreme learning machine-based pedestrian detection method," in *2013 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2013, pp. 1404–1409.

[8] M. Oren, C. Papageorgiou, P. Sinha, E. Osuna, and T. Poggio, "Pedestrian detection using wavelet templates," in *1997 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, 1997, pp. 193–199.

[9] B. Leibe, E. Seemann, and B. Schiele, "Pedestrian detection in crowded scenes," in *2005 EEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1. IEEE, 2005, pp. 878–885.

[10] D. Gavrila, "Pedestrian detection from a moving vehicle," *Computer Vision - ECCV 2000*, pp. 37–49, 2000.

[11] P. Sermanet, K. Kavukcuoglu, S. Chintala, and Y. LeCun, "Pedestrian detection with unsupervised multi-stage feature learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3626–3633.

[12] W. Ouyang and X. Wang, "Joint deep learning for pedestrian detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 2056–2063.

[13] Z. Cai, M. Saberian, and N. Vasconcelos, "Learning complexity-aware cascades for deep pedestrian detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 3361–3369.

[14] W. Ouyang and X. Wang, "A discriminative deep model for pedestrian detection with occlusion handling," *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3258–3265, 2012.

[15] Y. Tian, P. Luo, X. Wang, and X. Tang, "Pedestrian detection aided by deep learning semantic tasks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5079–5087.

[16] A. Angelova, A. Krizhevsky, V. Vanhoucke, A. S. Ogale, and D. Ferguson, "Real-time pedestrian detection with deep network cascades." *BMVC*, pp. 1–12, 2015.

[17] M. Pedersoli, J. Gonzalez, X. Hu, and X. Roca, "Toward real-time pedestrian detection based on a deformable template model," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 1, pp. 355–364, 2014.

[18] M. Hahnle, F. Saxen, M. Hisung, U. Brunsmann, and K. Doll, "Fpga-based real-time pedestrian detection on high-resolution images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2013, pp. 629–635.

[19] P. Dollar, C. Wojek, B. Schiele, and P. Perona, "Pedestrian Detection: An Evaluation of the State of the Art," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 4, pp. 743–761, 2012.

[20] V. Philomin, R. Duraiswami, and L. Davis, "Pedestrian tracking from a moving vehicle," in *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2000, pp. 350–355.

[21] R. M. Mueid, C. Ahmed, and M. A. R. Ahad, "Pedestrian activity classification using patterns of motion and histogram of oriented gradient," *Journal on Multimodal User Interfaces*, vol. 10, no. 4, pp. 299–305, 2016.

[22] F. Xu, X. Liu, and K. Fujimura, "Pedestrian detection and tracking with night vision," *IEEE Transactions on Intelligent Transportation Systems*, vol. 6, no. 1, pp. 63–71, 2005.

[23] G. Grubb, A. Zelinsky, L. Nilsson, and M. Rilbe, "3d vision sensing for improved pedestrian safety," in *2004 IEEE Intelligent Vehicles Symposium*. IEEE, 2004, pp. 19–24.

[24] C. Premebida, G. Monteiro, U. Nunes, and P. Peixoto, "A lidar and vision-based approach for pedestrian and vehicle detection and tracking," in *2007 IEEE Intelligent Transportation Systems Conference (ITSC)*. IEEE, 2007, pp. 1044–1049.

[25] A. Leykin and R. Hammoud, "Robust multi-pedestrian tracking in thermal-visible surveillance videos," in *2006 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW)*. IEEE, 2006, pp. 136–136.

[26] A. Ess, B. Leibe, K. Schindler, and L. Van Gool, "A mobile vision system for robust multi-person tracking," in *2008 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2008, pp. 1–8.

[27] D. Mitzel, E. Horbert, A. Ess, and B. Leibe, "Multi-person tracking with sparse detection and continuous segmentation," *Computer Vision–ECCV 2010*, pp. 397–410, 2010.

[28] A. J. Lipton, H. Fujiyoshi, and R. S. Patil, "Moving target classification and tracking from real-time video," in *Proceedings of the Fourth IEEE Workshop on Applications of Computer Vision (WACV)*. IEEE, 1998, pp. 8–14.

[29] B. Benfold and I. Reid, "Stable multi-target tracking in real-time surveillance video," in *2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2011, pp. 3457–3464.

[30] J. Ge, Y. Luo, and G. Tei, "Real-time pedestrian detection and tracking at nighttime for driver-assistance systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 10, no. 2, pp. 283–298, 2009.

[31] Y. Yang and D. Ramanan, "Articulated pose estimation with flexible mixtures-of-parts," in *2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2011, pp. 1385–1392.

[32] G. Shakhnarovich, P. A. Viola, and T. Darrell, "Fast pose estimation with parameter-sensitive hashing." *ICCV*, vol. 3, p. 750, 2003.

[33] C.-P. Lu, G. D. Hager, and E. Mjolsness, "Fast and globally convergent pose estimation from video images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 6, pp. 610–622, 2000.

[34] M. Andriluka, S. Roth, and B. Schiele, "Monocular 3d pose estimation and tracking by detection," in *2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2010, pp. 623–630.

[35] A. Jain, J. Tompson, Y. LeCun, and C. Bregler, "Modeep: A deep learning framework using motion features for human pose estimation," in *Asian Conference on Computer Vision*. Springer, 2014, pp. 302–315.

[36] J. J. Tompson, A. Jain, Y. LeCun, and C. Bregler, "Joint training of a convolutional network and a graphical model for human pose estimation," *Advances in neural information processing systems*, pp. 1799–1807, 2014.

[37] W. Ouyang, X. Chu, and X. Wang, "Multi-source deep learning for human pose estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2329–2336.

[38] A. Toshev and C. Szegedy, "Deeppose: Human pose estimation via deep neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1653–1660.

[39] M. Campbell, M. Egerstedt, J. P. How, and R. M. Murray, "Autonomous driving in urban environments: approaches, lessons and challenges," *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 368, no. 1928, pp. 4649–4672, 2010.

[40] C. Urmson, J. Anhalt, D. Bagnell, C. Baker, R. Bittner, M. Clark, J. Dolan, D. Duggins, T. Galatali, C. Geyer *et al.*, "Autonomous driving in urban environments: Boss and the urban challenge," *Journal of Field Robotics*, vol. 25, no. 8, pp. 425–466, 2008.

[41] Q. Yi, S. Chien, J. Brink, Y. Chen, L. Li, D. Good, C.-C. Chen, and R. Sherony, "Mannequin development for pedestrian pre-collision system evaluation," in *2014 IEEE 17th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2014, pp. 1626–1631.

[42] J. Ji, A. Khajepour, W. W. Melek, and Y. Huang, "Path planning and tracking for vehicle collision avoidance based on model predictive control with multi-constraints," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 2, pp. 952–964, 2017.

[43] H. Ahn, A. Rizzi, A. Colombo, and D. Del Vecchio, "Experimental testing of a semi-autonomous multi-vehicle collision avoidance algorithm at an intersection testbed," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2015, pp. 4834–4839.

[44] R. Fujishiro, C. Liu, L. Christopher, and J. Y. Zheng, "Bicyclist behavior analysis for pcs (pre-collision system) based on naturalistic driving," in *24th International Technical Conference on the Enhanced Safety of Vehicles (ESV)*, no. 15-0211, 2015.

[45] A. Ngre, C. Braillon, J. L. Crowley, and C. Laugier, *Real-Time Time-to-Collision from Variation of Intrinsic Scale*. Springer Berlin Heidelberg, 2008.

[46] G. P. Stein, O. Mano, and A. Shashua, "Vision-based ACC with a single camera: bounds on range and range rate accuracy," in *IEEE IV2003 Intelligent Vehicles Symposium. Proceedings (Cat. No.03TH8683)*, 2003, pp. 120–125.

[47] D. Siganos and C. Stergiou, "Neural networks and their uses," *Imperial College London: Surveys and Presentations in Information Systems Engineering*, 1996.